Remote Experiments as an Asset for Learning Programming in Python

H. Guerra [1,2], A. Cardoso [3], V. Sousa[3] and L.M. Gomes [1,2]

[1] University of Minho, Braga, Portugal
[2] University of Azores, Ponta Delgada, Portugal
[3] University of Coimbra, Coimbra, Portugal

Abstract—The online experimentation is in expansion in Higher Education Institutions, with a dynamic extension to all scientific areas. The integration of online labs in higher educational courses constitutes a significant financial saving way for the institutions with multiple benefits for student's learning. Without access and usage time restrictions, these labs provide a wide range of experiments, performed individually or in groups, repeated as often as needed, without any space or time restrictions, just imposing the Internet access is available and in good conditions. This paper describes a use case application of a web platform that integrates, monitors and controls different technologies, such as Wireless Sensor Networks, Virtual Sensor Networks and Data Acquisition Boards, through an Internet accessible RESTful API, using low-cost devices such as the Raspberry Pi. In this context, a remote laboratory is developed for students programming in Python, based on such platform to improve their programming skills, overcome their difficulties in CS1 and CS2 courses, and give the ability to interact to physical world through practical examples.

Index Terms—Online experimentation; programming in Python; remote labs; remote sensing; wireless sensor networks.

I. INTRODUCTION

Laboratory activities, fundamental for the development of skills related to problem solving, the linkage between theory and practice, experimentation and encouragement of creativity, can now benefit from the online labs ([1], [2], [3]).

The online experimentation is in expansion in Higher Education Institutions, with a dynamic extension to many scientific areas [4]. The integration of online labs in higher educational courses constitutes a significant financial saving way for the institutions with multiple benefits for student's learning. Without access and usage time restrictions, these labs provide a wide range of experiments, performed individually or in groups, repeated as often as needed, without any space or time restrictions, just imposing the Internet access is available and in good conditions.

During the first year, some Computer Science study cycles do not require CS1 (introduction to programming) course as a formal prerequisite for attending CS2 (basic data structures) course [5], allowing students to attend the latter without having the required knowledge. Besides the supplementary effort for both students and teachers in overcome it, academic results are not satisfactory. This problem has been subject of reflections and many contributions for the continuous improvement of the programming teaching/learning process (e.g., [6], [7] and [8]).

Many programming educational projects have been widely used in various approaches, contributing to promote programming skills in students and overcome their difficulties. Some of these projects include the adoption of interpreted languages, such as Python, to teach basic and intermediate programming courses for young people [9]. Taking advantage of a syntactic language simplification, these approaches intend to provide a more accessible and attractive learning way more focused on the problem solving rather than on programming language itself.

With the advent of Internet of Things, the actual approaches to develop programming skills, such as algorithm thinking and logical reasoning, in 1-year-computer science students, based on project-oriented learning processes, can be extended to use experimental setups with low cost microcomputers.

In this paper, we describe a use case application of a web platform that integrates, monitors and controls different technologies, such as Wireless Sensor Networks, Virtual Sensor Networks and Data Acquisition Boards, through an Internet accessible RESTful API, using low-cost devices such as the Raspberry Pi [10]. In this context, a remote laboratory is developed for students of a CS2 course of Informatics - Networks and Multimedia at the University of Azores programming in Python. The remote laboratory was integrated in the project component assessment of the course and it intended to improve the quality of academic projects, by providing students the ability to do remote experimentation with real systems, in real-time and with fewer access restrictions, and also contributed to improve their programming skills and overcome their difficulties in CS1 and CS2 courses. Comparing with the academic results obtained on the previous years, the remote lab contributed to a higher percentage of students enrolled in the projects and increased the percentage of positive grades.

II. OVERVIEW OF THE PROJECT

A. Students and course characterization

The experience took place during the second academic semester in 2015 (between February and June). The CS2 course aimed to provide basic knowledge on elementary data structures and algorithms, in the context of the object-oriented paradigm [11]. It has 6 ECTS, corresponding to a total of 168 working hours for students divided by 63 contact hours and 105 autonomous work hours.

77 students are initially enrolled in the course. Each topic of the syllabus is developed in the classroom, with several illustrative examples of the most important conceptual and technical aspects. A group project assessment component assumes relevance in the overall assessment of the course, allowing a supervised autonomous teamwork. This component is composed by several inter-related small projects developed during the semester and occupying the autonomous work hours, allowing a gradual application of knowledge. An agile methodology based on Scrum [12] is adopted to supervise the work of all student teams, trying to overcome the student's difficulties of learning as soon as possible and discouraging them to accumulate doubts. An agile methodology is also recommended for each team management. The development of each small project is phased, and may have multiple intermediate deliveries, and ends with a presentation and discussion meeting, resulting in a quantitative assessment of the project. The assessment included penalties for non-compliance of the intermediate deliverables and supplementary values for solutions that were not limited to the points outlined in the statement. Regrouping of students is allowed after the delivery of each project. It is intended to promote the creation of new teams with students who are subject to an overload of work due to withdrawal of the remaining group of colleagues, and that may discourage deeply.

B. Projects description

Projects involve remote interactions with a Web platform connected with a RaspberryPi, which is connected to a wireless sensor network (WSN) used for measure air temperature, humidity and luminosity, in distributed in different locations of in Lab of Industrial Informatics and Systems (LIIS) of the University of Coimbra. Data acquired from the sensors are sent through the WSN to the Raspberry Pi and are recorded with a given sampling time.

The experience had begun with a simple project, the shorter and less demanding. Its main goal was to evaluate students' knowledge at the level of a CS1 course, giving opportunity to students to overcome, as soon as possible, the difficulties that they should have, and thus reducing their risk of failure in the CS2 course. The main task for students was to develop a small Python application to display, for each location node and also for entire network, descriptive statistical information about the data collected by sensors. All data and results should be saved in CSV files for further use [13]. Students could run remotely their Python programmes on the RaspberryPi. The platform includes quizzes for self-assessment and forums to promote the clarification of questions on the use of the remote system and troubleshooting on programming.

The second project was the more critical project in the sense that its success would determine the success of the later project. The project intends to develop an application to support the management of greenhouses. Each greenhouse is connected to a WNS and has a set of control parameters for desirable environmental conditions (e.g., air temperature, humidity, luminosity) to operate without external intervention (e.g., opening/closing windows, turning on/off light). The application should provide to collect data in real time from WSN sensor and get information from several filters (e.g., greenhouses with certain products /certain products in certain states of maturation / intervention needs).

An extension of the second project to provide the management of greenhouse graphs and applies some graph algorithms (e.g., shortest path, all paths) and sorting algorithms based on ordered trees to obtain information from several filters, characterized the third project.

An implementation of a Remote Lab for Python Programming (RLPP) to simulate real conditions of air temperature, humidity, and luminosity from the interior of a house, such as a greenhouse, allowed, using an online platform, students could obtain data in real time, from a remote system.

III. THE REMOTE LAB

A Web platform that integrates, monitors and controls different technologies, such as Wireless Sensor Networks, Virtual Sensor Networks and Data Acquisition Boards, through an Internet accessible RESTful API, using low-cost devices such as the Raspberry Pi is applied to develop the RLLP, whose architecture is shown in Fig 1. The front-end was developed on a Model View Controller framework used to create the entire interface, including file submission and download, camera visualization and web socket communication to the back-end server that is running on a Raspberry Pi.

The activity to support project 1 described in the previous section consisted in building programs and run them remotely on a RP, which is connected to a WSN. The WSN has 3 sensor nodes distributed on different locations of LIIS that measure temperature, humidity and luminosity [13].

For this activity it was developed a Python module with a set of functions to fetch data from the WSN was developed so that the students only had to develop the core part of the program without worrying on understanding how the interaction with the platform is made. To the access to the remote laboratory, a management system was provided by the platform, based on a First Come, First Served policy, with the establishment of a maximum threshold for individual usage time. In Fig. 2 it is possible to observe the setup environment.

To carry out the project of a registered student, the procedure for testing the code involves the following steps: (1) Login to the platform; (2) Submission of a file with the program in Python; (3) Waiting for the turn to run the users code; (4) Program execution with visualization, in real time, of the results shown in a Raspberry Pi monitor; (5) Download of a file with data recorded.

After a user's program execution and if it did generate any file, it is always available through the main menu of the platform. A sample test file is shown if Fig. 3.

For projects 2 and 3, the interaction with the platform is made through a RESTful API that has endpoints to fetch recently collected data, fetch data that was previously sto-
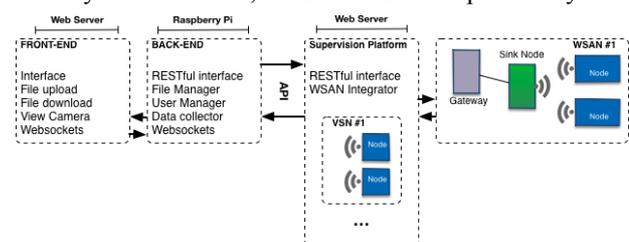

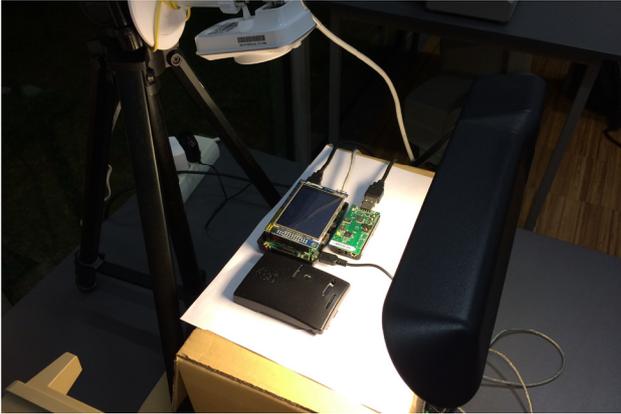Figure 1. Architecture of the Remote Laboratory for Programming in Python.

Figure 2.   Photo describing set up environment.

```
# All the users must import this functions
from nim import startup, getdata, closeup, savedata

# startup('192.168.1.5','8082')
startup()

print 'Collecting data...'
n = {}
for i in range(1, 4):
    valT = []
    valH = []
    valL = []
    for x in range(10):
        triple = getdata(i)
        valT.append(triple[0])
        valH.append(triple[1])
        valL.append(triple[2])

        n[i] = [min(valT), max(valT), sum(valT) / len(valT),
                min(valH), max(valH), sum(valH) / len(valH),
                min(valL), max(valL), sum(valL) / len(valL),
                valT, valH, valL]

closeup()

# plotdata(n)
savedata(n)

print 'Media da Temperatura ' + str(sum(valT) / len(valT)) + ' C'
print 'Media da Humidade ' + str(sum(valH) / len(valH)) + ' %'
print 'Media da Luminosidade ' + str(sum(valL) / len(valL)) + ' Lux'
```

Figure 3.   Sample file test to access the web platform.

red on a database, fetch information about a given network and fetch all the available networks, virtual or real. It was implemented 9 virtual WSN with the same behavior of the real WSN.

IV. RESULTS

Comparing with the results obtained on the previous years, this remote lab contributed to a higher percentage of students enrolled in the projects and an increase of the percentage of positive grades. To be more specific, in the previous academic year, 2013/2014, the course had a total of 23 students that completed all the evaluation processes and this academic year, 2014/2015, there was a total of 32 students. The number of registrations on all of the projects in 2013/2014 was 31,08% of 74 students and this year the numbers raised to 41,56% of 77 students. Once again, this validates the use of real and interactive experiments contributes to a high-quality learning experience and motivates students to participate on the projects.

All of the interactions with the application of RLPP for project 1 were stored and counted for statistical purposes. During the group project, the total number of requests was 8933 and of those, 2370 were file uploads 520 file downloads. As we can see in figure 7.7, of the total number of uploads, there was a success rate of 72% and 28% of the submissions had errors.

REFERENCES

[1] de Jong, T.; Linn, M.; & Zacharia, Z. (2013). "Physical and virtual laboratories" in Science and Engineering Education. Science, 340 (6130), 305-308 http://dx.doi.org/10.1126/science.1230579

[2] Froyd, J. E.; Wankat, P. C.; & Smith, K. A. (2012). "Five major shifts in 100 years of Engineering Education". In Proceedings of the IEEE, Volume 100 (pp. 1344-1360) http://dx.doi.org/10.1109/jproc.2012.2190167

[3] Restivo, M.T. & Cardoso, A. (2013). "Promoting Remote and Virtual labs" in Portugal, International Journal of Engineering Pedagogy, 3, 11-15, 2013 http://dx.doi.org/10.3991/ijep.v3is1.2399

[4] Johnson, L., Adams Becker, S., Estrada, V., and Freeman, A. (2015). NMC Horizon Report: 2015 Higher Education Edition. Austin, Texas: The New Media Consortium

[5] Computer Science Curriculum 2013: Curriculum Guidelines for Undergraduate Degree Programs in Computer Science. ACM & IEEE., 2013. http://www.acm.org/education/CS2013-final-report.pdf.

[6] A. Gomes and A. J. Mendes, "Learning to program - difficulties and solutions". International Conference on Engineering Education ICEE, 2007. http://www.ineer.org/Events/ICEE2007/papers/411.pdf.

[7] C. Watson and F. Li, "Failure Rates in Introductory Programming Revisited", ACM Innovation and Technology in Computer Science Education (ITiCSE), pp.39-44, Jun. 2014

[8] A. Jayal, S. Lauria, A. Tucker and S. Swift, "Python for Teaching Introductory Programming: A Quantitative Evaluation". Innovation in Teaching and Learning in Information and Computer Sciences 10(1), 86-90, 2011. http://dx.doi.org/10.11120/ital.2011.10010086

[9] A. Bogdanchikov, M. Zhaparov and R. Suliyev, "Python to learn programming". Journal of Physics: Conference Series 423, 2013. http://iopscience.iop.org/1742-6596/423/1/012027/pdf/1742-6596_423_1_012027.pdf http://dx.doi.org/10.1088/1742-6596/423/1/012027

[10] V. Sousa (2015), Platform for the supervision of remote systems using low cost devices, Master Thesis in Informatics Engineering, Laboratory of Industrial Informatics and Systems, University of Coimbra

[11] K. Lambert, "Fundamentals of Python: Data Structures", Cengage Learning, 2013

[12] A. Pham and P.-V. Pham (2011), Scrum in Action, Cengage Learning, 1st Edition

[13] H. Guerra, A. Cardoso, V. Sousa, J. Leitão, V. Graveto, and L. Gomes (2015). "Demonstration of Programming in Python using a remote lab with Raspberry Pi". Proceedings of the 3rd Experiment@ International Conference – exp.at'15. pp. 101-102.

AUTHORS

H. Guerra is with the University of Azores, Rua Mãe de Deus, 9500-801 Ponta Delgada, Portugal, and with the Centre ALGORITMI of the University of Minho, Braga, Portugal (e-mail: helia@uac.pt).

A. Cardoso is with the Dep. of Informatics Engineering of the University of Coimbra, DEI, Polo II – UC, 3030-290 Coimbra, Portugal (email: alberto@dei.uc.pt).

V. Sousa is researcher at CISUC, Dep. of Informatics Engineering, Polo II – UC, 3030-290 Coimbra, Portugal (email: vhsousa@student.dei.uc.pt).

L. Gomes is with the University of Azores, Rua Mae de Deus, 9500-801 Ponta Delgada, Portugal, and with the Centre ALGORITMI of the University of Minho, Braga, Portugal (e-mail: lmg@uac.pt).